Solving the neutron transport equation using the radial basis function augmented with finite difference method[*]

Daming Yuan,[1] Zhanheng Cheng (Corresponding author),[2, †] and Zhen Huang[3]

[1]*School of Mathematics and Statistics, Jiangxi Normal University & Jiangxi Provincial Center of Applied Mathematics, 330095, China*
[2]*School of Mathematics and Statistics, Yili Normal University,*
Yili 835000, China, 8618099522106, Czh618czh@163.com
[3]*School of Mathematics and Statistics, Jiangxi Normal University, Nanchang 330095, China*

Given its high dimensionality and the intricate geometric shapes encountered in real-world applications, developing efficient numerical methods to solve the neutron transport equation is of considerable practical significance. Meshless methods offer a robust alternative to traditional numerical techniques, delivering notable advantages in flexibility, accuracy, and computational efficiency. This paper investigates the application of a radial basis function (RBF) method, combined with the finite-difference approach, to solve the two-dimensional steady-state one-group isotropic neutron transport equation. Unlike conventional numerical methods, the proposed approach enables rapid generation of data nodes without the need for predefined meshes, making it particularly effective for solving differential equations in complex geometries. The method employs two types of radial basis functions as interpolation bases, with the numerical solution obtained by determining the corresponding weights on a stencil of a defined size. To assess the effectiveness of the proposed method, numerical examples are provided that demonstrate its performance and accuracy.

Keywords: radial basis function, neutron transport equation, shape parameter, LOOCV

I. INTRODUCTION

The neutron transport equations describe how neutrons behave and distribute themselves within a given medium, accounting for phenomena like scattering, absorption, and external sources [1, 2]. These equations are foundational in nuclear science and engineering, with a wide range of applications across various fields. They are essential for predicting neutron flux, distribution, and reaction rates in materials, making them critical in nuclear reactor design [3, 4]. In medical fields, such as neutron capture therapy, these equations are used to effectively target tumors. Furthermore, they are crucial for ensuring adherence to nuclear safety regulations and security, and for evaluating nuclear criticality safety when dealing with fissile materials [5, 6]. In astrophysics and cosmology, neutron transport plays a key role in modeling nucleosynthesis processes in stars and supernovae [7, 8]. Additionally, these equations are vital for radiation shielding and the development of neutron detection systems [9, 10]. In fusion energy research [11–13], neutron transport is used to model the behavior of neutrons generated during fusion reactions. Lastly, these equations are applied to predict material property changes due to neutron irradiation, helping to understand the long-term effects of neutron exposure on materials [13–15].

Neutron transport problems can involve three spatial dimensions, along with temporal and angular dependencies. These equations often consider multiple energy groups as well. Additionally, neutron transport issues commonly occur in complex geometries, where the arrangement and shape of materials have a significant impact on neutron behavior. The high dimensionality and the complexity of these geometries present challenges for both analytical and numerical solutions to the equations.

Numerical methods for solving neutron transport equations are essential tools in nuclear science and engineering, facilitating the study of neutron behavior in various contexts. Common approaches include deterministic techniques like the finite difference method (FDM) [3, 16], continuous and discontinuous finite element methods (FEM) [17–19], the finite volume method (FVM) [20], and the stochastic Monte Carlo simulation method [21, 22]. Each method offers distinct advantages depending on the problem's complexity and the desired accuracy. These methods allow researchers to model neutron interactions with precision, optimize system designs, and ensure the safety of nuclear applications. As computational power continues to improve, these techniques evolve, providing deeper understanding of neutron transport phenomena.

Meshless methods, also referred to as mesh-free methods, are advanced numerical techniques that bypass the need for traditional mesh generation when solving partial differential equations. Instead of relying on a structured grid, these methods use a set of scattered points to represent the problem domain. This approach offers enhanced flexibility in dealing with complex geometries and enables adaptive refinement, making it particularly effective for problems with irregular shapes or evolving domains. Meshless methods have emerged as a promising and innovative approach for solving neutron transport equations, offering significant advantages in flexibility and accuracy [23–25].

II. RBF-FD METHOD

A. Radial basis functions

A radial basis function (RBF) is a type of function, which depends on the distance from a central point and is symmetric with respect to this point.

For a d-dimensional RBF $\phi : \mathbb{R}^d \rightarrow \mathbb{R}$, its variables are nonnegative and defined by $r = \|\mathbf{x} - \mathbf{x}_k\|_2$, where $\|\cdot\|_2$ represents the Euclidean distance, and $\mathbf{x}_k$ and $\mathbf{x}$ denote the center point and the assignment point, respectively. The most commonly used RBFs are listed in Table (1). Based on their smoothness, RBFs can be classified into two categories: (i) lightly segmented RBFs, such as the polynomial spline (PHS), which exhibit limited smoothness, and (ii) infinitely smooth RBFs, such as the Multiquadric (MQ), Guassian (GA), Inverse Multiquadric (IMQ), and Inverse Quadratic (IQ), which are highly differentiable and continuous at all orders [26].

Common RBF	$\phi(r = \|\mathbf{x} - \mathbf{x}_k\|_2)$
Polyharmonic Splines (PHS)	$r^{2m} \log r, m \in \mathbf{N}$
	$r^{2m+1}, m \in \mathbf{N^0}$
Multiquadric (MQ)	$\sqrt{1 + (\varepsilon r)^2}$
Gaussian (GA)	$e^{(\varepsilon r)^2}$
Inverse Multiquaric (IMQ)	$1/\sqrt{1 + (\varepsilon r)^2}$
Inverse Quarat (IQ)	$1/(1 + (\varepsilon r)^2)$

Table 1. List of common RBFs

Infinitely smooth radial basis functions (RBFs) typically involve one or more parameters that control their behavior, such as smoothness, scale, or influence width. These parameters are essential for customizing the RBFs to specific applications, including interpolation, approximation, and numerical methods.

The shape parameters can be divided into two categories: fixed parameters, which are independent of the nodes in the computational domain, and variable parameters, which depend on the node distribution to some extent. The fixed parameter approach is simpler but may reduce the overall accuracy of the solution. A comparison of these two types of shape parameters was conducted in [27], where the optimal values for each were determined. The study demonstrated that the choice of optimal shape parameters is influenced by factors such as the node distribution, the geometry of the computational domain, boundary conditions, and the governing equations of the problem. As the shape parameter decreases, the RBF becomes increasingly flattened, which leads to an ill-conditioned matrix. Consequently, there exists a specific range of shape parameters that ensures the RBF-FD (Finite Difference) method provides higher accuracy compared to traditional FD methods and minimizes truncation errors, leading to optimal results [28].

The RBF augmented with a finite difference method (RBF-FD), introduced in the early 21st century, is a powerful numerical approach for solving partial differential equations by integrating radial basis interpolation with the finite difference method [27–29]. The core concept of the RBF-FD is to approximate a differential operator as a linear combination of radial basis functions and polynomials in the vicinity of a central point. In comparison to traditional methods such as the finite difference method (FDM), finite element method (FEM), finite volume method (FVM), and spectral methods [30], the RBF-FD offers notable advantages, including enhanced geometric flexibility, high accuracy, and exceptional computational efficiency.

In this study, we select the Multiquadric (MQ) and Gaussian radial basis functions for interpolation, resulting in the MQ-FD and Gauss-FD algorithms, respectively. To investigate how the shape parameters of these two basis functions influence the accuracy and stability of the numerical solutions, we refer readers to the works in [27, 31]. The leave-one-out cross-validation (LOOCV) method, discussed in subsection (II C), will be employed for this analysis.

B. RBF-FD differentiation weight calculation

For a node $\{\mathbf{x}_k\}$, where $k = 1, \ldots, N$, in a scattered set of nodes $\{\mathbf{x}_k\}_{k=1}^{N}$, a stencil is constructed by selecting $n - 1$ surrounding nodes based on a specified rule. This stencil is denoted as $\mathcal{T}_k$, with $\{\mathbf{x}_k\}$ representing the center of the stencil and n being the stencil size. The RBF-FD method for solving differential equations operates on a principle similar to that of the classical finite difference method [32–35]. Specifically, by taking a linear combination of the radial basis functions ψ_i corresponding to the values at the n nodes in the stencil $\mathcal{T}_k$, the value of the differential operator $\mathcal{L}\psi(\mathbf{x}_k)$ can be approximated as

$$(\mathcal{L}\psi(\mathbf{x})))|_{\mathbf{x}=\mathbf{x}_k} = \sum_{i=1}^{n} \omega_i \psi_i. \tag{1}$$

The combination coefficients ω_i (for $i = 1, \ldots, n$) are referred to as the differential weights of the RBF-FD method. These weights can be determined by solving the following system of linear equations:

$$\mathbf{A}\omega = \mathbf{f}. \tag{2}$$

where,

$$\mathbf{A} = \begin{bmatrix} \phi(\|\mathbf{x}_1 - \mathbf{x}_1\|) & \cdots & \phi(\|\mathbf{x}_1 - \mathbf{x}_n\|) \\ \vdots & \ddots & \vdots \\ \phi(\|\mathbf{x}_n - \mathbf{x}_1\|) & \cdots & \phi(\|\mathbf{x}_n - \mathbf{x}_n\|) \end{bmatrix},$$
$$\omega = \begin{bmatrix} \omega_1, \cdots, \omega_n \end{bmatrix}^T,$$
$$\mathbf{f} = \begin{bmatrix} \mathcal{L}\phi(\|\mathbf{x} - \mathbf{x}_1\|)|_{\mathbf{x}_c} \cdots \mathcal{L}\phi(\|\mathbf{x} - \mathbf{x}_n\|)|_{\mathbf{x}_c} \end{bmatrix}^T. \tag{3}$$

As the stencil size n increases, the matrix $\mathbf{A}$ tends to become ill-conditioned. To mitigate this issue, the RBF-FD method assumes that the interpolation function $s(\mathbf{x})$ is a linear combination of the radial basis function (RBF) and an associated polynomial. For instance, an interpolation function that includes a linear polynomial in a two-dimensional space

can be expressed as:

$$\psi(\mathbf{x}) = \sum_{i=1}^{n} \omega_i \phi(\|\mathbf{x} - \mathbf{x}_i\|) + \gamma_1 + \gamma_2 x + \gamma_3 y. \quad (4)$$

To ensure the system is well-posed, the weight vector ω must satisfy the following constraint:

$$\sum_{i=1}^{n} \omega_i = \sum_{i=1}^{n} \omega_i x_i = \sum_{i=1}^{n} \omega_i y_i = 0.$$

Theoretically, regardless of the node configuration, as long as the node set is unisolvent with respect to the polynomial, the inclusion of polynomials in radial basis interpolation guarantees that the interpolation problem remains unisolvent [33].

Furthermore, the weight vector ω and the polynomial coefficient vector γ can be determined by solving the following linear system:

$$\begin{bmatrix} \mathbf{A} & P \\ P^T & 0 \end{bmatrix} \begin{bmatrix} \boldsymbol{\omega} \\ \boldsymbol{\gamma} \end{bmatrix} = \begin{bmatrix} \mathbf{f} \\ \mathcal{L}\boldsymbol{p} \end{bmatrix}, \quad (5)$$

where the matrices $\mathbf{A}$ and $\mathbf{f}$ are defined in (3), and the matrices $\mathbf{P}$ and $\mathcal{L}\boldsymbol{p}$ are given by:

$$P = \begin{bmatrix} 1 & x_1 & y_1 \\ \vdots & \vdots & \vdots \\ 1 & x_n & y_n \end{bmatrix},$$

$$\mathcal{L}\boldsymbol{p} = \begin{bmatrix} \mathcal{L}1|_{\mathbf{x}_c}, \mathcal{L}x|_{\mathbf{x}_c}, \mathcal{L}y|_{\mathbf{x}_c} \end{bmatrix}^T,$$

C. Optimal shape parameter selection by Leave-One-Out method

The leave-one-out cross-validation (LOOCV) method [27] is employed to determine the optimal shape parameters for each stencil $\mathcal{T}_k$.

LOOCV is a type of cross-validation technique commonly used to evaluate the performance of a model, especially when the dataset is small. The method works by iteratively leaving out one data point from the training set, using the remaining points to train the model, and then testing the model on the left-out point. This process is repeated for every data point in the dataset, with the overall performance being averaged. The key steps of the method are outlined as follows:

1. Leave out one data point x_k and obtain the remaining $n-1$ points:

$$\mathbf{x}^{[k]} = [\mathbf{x}_1, \ldots, \mathbf{x}_{k-1}, \mathbf{x}_{k+1}, \ldots, \mathbf{x}_n]^T.$$

Here, the superscript $[k]$ indicates that the k-th data point, x_k, is left out.

2. Train the model using the data vector $\mathbf{x}^{[k]}$.

3. Test the trained model from step 2 on the left-out data point x_k.

4. Compute the prediction error for the left-out point x_k.

5. Repeat steps 1-4 for all data points in the dataset $\mathcal{T}_k$.

6. Evaluate the overall performance of the model by calculating the average of all prediction errors.

The outcome of the LOOCV method is an error vector, which is computed using the ℓ_2 norm to define the cost function for determining the optimal value of the shape parameter ϵ. Since the LOOCV method relies on errors derived from the given data, the predicted optimal shape parameters are typically close to the true optimal values [27].

When extending the LOOCV method to RBF-FD, we define the vectors as follows: $\vec{\phi}^{[k]}$ and $\lambda^{[k]}$ as follows.

$$\begin{aligned} \vec{\phi}^{[k]} &= [\phi_1, \cdots, \phi_{k-1}, \phi_{k+1}, \cdots, \phi_n], \\ \lambda^{[k]} &= [\lambda_1, \cdots, \lambda_{k-1}, \lambda_{k+1}, \cdots, \lambda_n]. \end{aligned}$$

where $\phi_i = \phi(\| \mathbf{x} - \mathbf{x}_i \|)$. The interpolations (4) are approximated by

$$P_\phi^{[k]} = \sum_{j=1}^{n-1} \left(\lambda_j^k \phi\left(\| \mathbf{x} - \mathbf{x}_j^{[k]} \|\right) + 1 + \mathbf{x}_j^{[k]} + \mathbf{y}_j^{[k]} \right),$$

and the error is defined by:

$$e_k = \psi(\mathbf{x}_k) - P^{[k]}(\mathbf{x}_k).$$

The effectiveness of the fitting to the dataset is assessed using the ℓ_2 norm of the error vector $\mathbf{e} = [e_1, \ldots, e_n]^T$. This norm serves as the cost function for determining the optimal value of the shape parameter ϵ. The Matlab function `fminbnd` can be employed to find the optimal value of ϵ.

It is important to note that the exact solution cannot be directly applied to obtain the function values $\vec{\psi}^{[k]}$ needed for interpolation. Typically, the shape parameter must be known a priori in order to solve the PDE and obtain a set of numerical solutions $\vec{\psi}^{[k]}$. This dataset of function values is then used in the LOOCV method for error evaluation.

III. NEUTRON TRANSPORT EQUATION AND NUMERICAL METHODS

A. Neutron transport equation

In $\mathcal{D} \subset \mathbb{R}^2$, the steady-state one-group neutron transport equation can be written as an integer-differential equation given by:

$$\begin{aligned} \Omega \cdot \nabla \psi(\mathbf{x}, \Omega) + \sigma_t \psi(\mathbf{x}, \Omega) = \\ \frac{1}{2} \left[\sigma_s \int_{\mathbb{S}^2} f(\Omega, \Omega') \psi(\mathbf{x}, \Omega') \, d\Omega' \right] + Q(\mathbf{x}) \end{aligned} \quad (6)$$

where $\psi(\mathbf{x}, \Omega)$ denotes the neutron angular flux at a position $\mathbf{x}$ and in a unit direction Ω; σ_t and σ_s represent the

total cross-section and the scattering cross-section, respectively; $Q(\mathbf{x}, \Omega)$ represents the internal fission rate within $\mathcal{D}$ or an external source term; and $\mathbb{S}^2$ denotes the unit circle in the two-dimensional space [1].

The scalar flux is another crucial quantity that characterizes the neutron transport equation, and it is defined as:

$$\Phi(\mathbf{x}) = \int_{\mathbb{S}^2} \psi(\mathbf{x}, \Omega') \mathrm{d}\Omega'. \tag{7}$$

The scattering function f is position-independent and represents the probability that a particle's direction changes to Ω after scattering from an initial direction Ω'; it also satisfies the following condition:

$$\int_{\mathbb{S}^2} f(\Omega, \Omega') \, \mathrm{d}\Omega' = 1. \tag{8}$$

For Eq. (6) to be well-posed, it must be accompanied by the appropriate boundary conditions; additionally, for time-dependent problems, suitable initial conditions are required. A detailed discussion of common boundary conditions for transport equations can be found in [1, 36, 37].

For clarity, the above equation can be rewritten as follows:

$$L\psi = S\psi + \frac{Q(\mathbf{x})}{2}, \tag{9}$$

where L is the leakage operator and S is the scattering operator, which are defined as follows:

$$L = \nabla + \sigma_t, \quad S = \frac{\sigma_s}{2} \int_{\mathbb{S}^2} f(\Omega, \Omega') \cdot \mathrm{d}\Omega'.$$

The above expressions describe the leakage and collision processes during transport, as well as the scattering process [1].

B. Source iteration method

The Source Iteration (SI) method is the most commonly used approach for numerically solving transport equations, and its iteration scheme is as follows:

$$\begin{aligned} \Omega \cdot \nabla \psi^{(l+1)} + \sigma_t \psi^{(l+1)} = \\ \frac{\sigma_s}{2} \int_{\mathbb{S}^2} f(\Omega, \Omega') \psi^{(l)}\left(\mathbf{x}, \Omega'\right) \mathrm{d}\Omega' + \frac{Q(\mathbf{x}, \Omega)}{2}, \end{aligned} \tag{10}$$

or

$$L\psi^{(l+1)} = S\psi^{(l)} + \frac{Q(\mathbf{x}, \Omega)}{2}. \tag{11}$$

The initial value of $\Phi^{(0)}$ in the iteration is typically set to zero. Physically, the Source Iteration (SI) method represents the angular flux after l collisions. The l-th scalar flux $\Phi^{(l)}(x)$ is then substituted into the right-hand side of Eq. (10) iteratively until the norm of the scalar flux difference between consecutive iterations satisfies the convergence criterion, i.e., $\| \Phi^{(l+1)}(x) - \Phi^{(l)}(x) \| \leq \varepsilon$, where ε is the specified termination tolerance.

C. Discrete ordinate method

When solving the neutron transport equation numerically, it is necessary to discretize both the spatial and angular variables. Common discretization methods for the angular variable include the discrete ordinates method (S_n) and spherical harmonics (P_n). The S_n method discretizes the angular variable into a set of direction vectors as required and solves the iterative equation for each direction [1, 38]. Compared to the P_n method, the S_n method is simpler to implement but is more susceptible to ray effects [1]. In this study, the discrete ordinates method and numerical integration are used to approximate the integral operator.

For the neutron transport equation in two-dimensional space, as given by Eq. (10), the direction Ω is discretized as $\Omega = (\cos\theta, \sin\theta)$, where θ represents the scattering angle relative to the position $\mathbf{x}$. The operator L is then expressed as follows:

$$L = \cos\theta \cdot \frac{\partial}{\partial_x} + \sin\theta \cdot \frac{\partial}{\partial_y} + \sigma_t. \tag{12}$$

Eq. (10) can then be expressed as:

$$L\psi^{(l+1)}(\mathbf{x}, \theta) = S \int_{-1}^{+1} \psi^{(l)}(\mathbf{x}, \mu') \mathrm{d}\mu' + \frac{Q(\mathbf{x})}{2}, \tag{13}$$

where $\mu = \cos\theta$ and $\mu' = \cos\theta'$.

In this study, the Henyey-Greenstein scattering function [1] is used to model the angular distribution of scattered neutrons. This function represents the angular probability distribution of scattered particles as a function of the cosine of the scattering angle, $\mu = \cos\theta$, and is defined as follows:

$$f(\mu) = \frac{1}{2\pi} \frac{1 - g^2}{(1 + g^2 - 2g\mu)^{3/2}}, \tag{14}$$

where g represents the average cosine of the scattering angle, also known as the anisotropy factor, which quantifies the directional bias of the scattering. The value of g determines the degree of anisotropy in the scattering distribution, as follows:

- $g = 0$ indicates isotropic scattering;

- $g > 0$ indicates forward scattering;

- $g < 0$ indicates backward scattering.

Furthermore, M discrete directions, denoted by $\theta_m = (m-1)\Delta\theta$ for $m = 1, \ldots, M$, are selected, where $\Delta\theta = \frac{2\pi}{M}$. The weight $w_{mm'}$, with $m, m' = 1, \ldots, M$, is defined as follows:

$$w_{mm'} = \int_{\theta_{m'}-\Delta\theta/2}^{\theta_{m'}+\Delta\theta/2} f(\theta_m, \theta') \, \mathrm{d}\theta'. \tag{15}$$

In addition, the sum of weights $\{w_{m,m'}\}$ is approximately equal to two, which can be expressed by:

$$\sum_{m'=1}^{M} w_{mm'} \approx 2, \ m = 1, ..., M. \qquad (16)$$

Next, $w_{mm'}$ and the angle discrete point θ_m are used to integrate the scattering term numerically, and the right end of Eq. 10) can be expressed as:

$$J^{(l)}(\mathbf{x}) = \sigma_s \sum_{m'=1}^{M} w_{mm'} \psi^{(l)}(\mathbf{x}, \mu_{m'}) + Q(\mathbf{x}). \qquad (17)$$

Correspondingly, the S_n method's source iteration process, defined by Eq. 10 or Eq. 11), can be expressed as follows:

$$L\psi_m^{(l+1)}(\mathbf{x}) = J_m^{(l)}, \ m = 1, \cdots, M, \qquad (18)$$

where $\psi_m = \psi(\mathbf{x}, \theta_m)$ represents the continuous angular flux solution in a discrete direction θ_m. It can be seen from the above that the source iteration method does not require solving the coupled differential equations in space and can avoid the inversion of large matrices.

In the SI method (18), it is needed to integrate the scattering term $J_m^{(l)}$ numerically and solve the hyperbolic partial differential equation (18) in each discrete direction θ_m. Because each direction of the source iteration is performed independently of the other directions, the computational cost increases linearly with the number of directions used in the numerical format [1].

IV. SOLVING NEUTRON TRANSPORT EQUATIONS BY RBF-FD METHOD

According to the operational principle of the Radial Basis Function-based Finite Difference (RBF-FD) method, for a given stencil $\mathcal{T}_k$ where $k = 1, \ldots, N$, the differential operator $\mathcal{L}_m = \Omega_m \cdot \nabla$ (with $m = 1, \ldots, M$) is applied in the mth direction. The corresponding weight vector

$$[\omega_{m,k}^{(1)}, \omega_{m,k}^{(2)}, \ldots, \omega_{m,k}^{(n)}], \quad m = 1, \ldots, M, \quad k = 1, \ldots, N,$$

is computed using Eq. (5). Here, $\omega_{m,k}^{(i)}$ for $i = 1, \ldots, n$ represents the i-th element of the k-th row of the $N \times N$ matrix A_m, with its position in the k-th row corresponding to the node's global index.

Next, we define the diagonal matrix

$$\mathbf{I}_m = \mathrm{diag}(\sigma_{t,1}, \ldots, \sigma_{t,N})$$

and the vector

$$\mathbf{J}^{(l)} = \left[J^{(1)}(\mathbf{x}_1), \ldots, J^{(l)}(\mathbf{x}_N) \right],$$

and solve the following linear system:

$$(A_m + \mathbf{I}_m)\psi_m^{(l+1)} = \mathbf{J}^{(l)}, \qquad \text{(July.01.01)}$$

to obtain the iterative angular flux $\psi_m^{(l+1)}$ in the mth direction of the S_N method.

A. Node and stencil layout

To investigate the convergence of meshless methods, as discussed in [32, 33], the mesh ratio ρ_X for a set of N scattered data points $X \subset \mathbb{R}^d$ is defined as

$$\rho_X = \frac{h_{x,\Omega}}{q_X} \geq 1.$$

In this context, the fill distance $h_{x,\Omega}$ and the separation distance q_X are given by

$$h_{x,\Omega} := h_X = \sup_{x \in \Omega} \min_{x_j \in X} \|x - x_j\|_{\ell_2(\mathbb{R}^d)},$$

and

$$q_X := \frac{1}{2} \min_{i \neq j} \|x_i - x_j\|_{\ell_2(\mathbb{R}^d)},$$

respectively.

In the RBF-FD method, for a quasi-uniform node distribution, the convergence order is given by $\mathcal{O}(h_X^{p-k+1})$, where k is the order of the differential operator being approximated, p is the degree of the appended polynomial, and the radial basis function used is the polyharmonic spline [32, 33]. For highly non-uniform node distributions with a large mesh ratio, the asymptotic convergence rate remains of the same order but is accompanied by a large leading constant, which diminishes the final precision in the error bound [32, 33]. For simplicity, this study adopts a Cartesian node layout to represent the node distribution and to demonstrate how to select the appropriate stencil based on the specific direction.
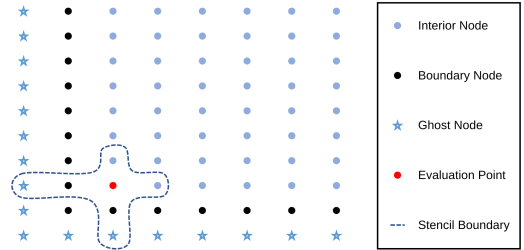

Fig. 1. Biased stencil and nodal distribution

Near the boundary of a physical region, the use of one-sided stencils can often lead to Runge's phenomenon. When solving elliptic equations, ghost points (i.e., virtual nodes) are typically located outside the physical domain, as illustrated in Fig. (1). These ghost points extend the layout pattern of the interior nodes. In general, the center point of a stencil that includes ghost points is referred to as an irregular interior point; otherwise, it is considered a regular interior point. In [33, 36], three methods for defining constraints on ghost points are discussed. Once the ghost points are determined, the function values at these locations can be obtained by imposing additional constraints on the boundary nodes. Similarly, the PDE can be solved on the boundaries, providing further constraints.

Considering the previously discussed method S_n, for a given direction vector and center point, this study determines the corresponding deflection stencil using the windward technique. Specifically, the center point will have a different number of nodes on the left and right or the upper and lower sides in two parallel coordinate directions. As illustrated in Fig. (2), the stencils can be classified into two categories as follows:

- If the angle between the direction vector Ω and a coordinate axis is non-zero, the number of nodes selected in the windward direction along the x- and y-axes will be greater than the number of nodes in the upwind direction;

- If the angle between the direction vector Ω and a coordinate axis is zero, the number of nodes selected in the windward direction along the corresponding coordinate axis will be greater than the number of nodes in the upwind direction. In the direction of the other coordinate axis, the number of nodes will be chosen symmetrically.
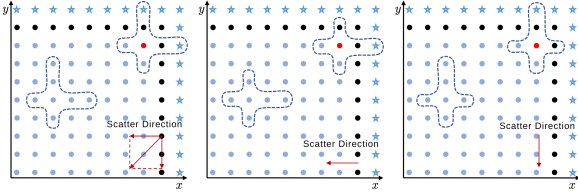

Fig. 2. Chose nodal points in upwind directions depends on whether the scattering directions parallels to the coordinate.

The number of degrees of freedom in the RBF-FD method corresponds to the value of the function $\psi(x)$ at each node. Therefore, in addition to the two types of local node stencils mentioned earlier, it is necessary to define the global numbering of the stencil nodes associated with the scattering direction to solve Eq. (6). For a two-dimensional space, the global numbering of the interior nodes is presented in four distinct forms, based on the cosine of the angle between the scattering direction vector Ω and the coordinate axes (both positive and negative). These configurations are illustrated in Fig. (4).

B. Results of the numerical experiments

The proposed method was validated through numerical experiments. Specifically, the algorithm was applied to solve the steady-state two-dimensional neutron transport equation. To assess the impact of directional discretization on the accuracy of the numerical results, the S_{64} discrete direction vector Ω was selected for the simulation, as the direction dissection number in the S_N method has a minimal effect on the accuracy. The physical domain for the equation was defined as $\mathcal{D} = [-10, 10] \times [-10, 10]$. The source term function was specified as:

$$Q(x, y, \theta) = \sigma_a e^{-4\sigma_a(x^2+y^2)} \left(1 - 8x\cos\theta - 8y\sin\theta\right).$$

At the incident boundary $\Gamma_{in} = \{\partial\mathcal{D}\times\mathbb{S}^2 \mid \boldsymbol{n}(x_b, y_b)\cdot\theta < 0\}$, the boundary condition was given by:

$$\psi(x_b, y_b, \theta) = e^{-4\sigma_a\left(x_b^2+y_b^2\right)}.$$

The total cross-section and absorption cross-section were specified as $\sigma_t = 1\,\mathrm{mm}^{-1}$ and $\sigma_a = 0.01\,\mathrm{mm}^{-1}$, respectively, while the average cosine value of the scattering angle was $g = 0.9$. Using these parameters, the isotropic exact solution to the two-dimensional steady-state transport equation was obtained as:

$$\psi_e(x, y, \theta) = e^{-4\sigma_a(x^2+y^2)},$$
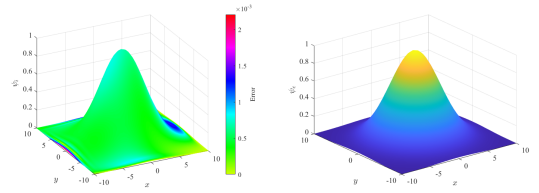
where $\forall\theta \in [0, 2\pi]$.

Fig. 3. Comparison between numerical solution with exact solution when the mesh is 41×41.

In the numerical solution process, $N = N_x \times N_y$ RBF center nodes were used, arranged in a Cartesian coordinate system. For simplicity, the same set of nodes was utilized as both the interpolation centers and the assignment points.

The maximum and average errors between the exact solution ψ_e and the numerical solution ψ were computed using the following definitions:

1. The maximum error is given by:

$$\epsilon_{\max} = \max_{i,j} |\psi(x_j, y_j, \theta_i) - \psi_e(x_j, y_j, \theta_i)|.$$

2. The mean square error (MSE) is defined as:

$$\epsilon_{\mathrm{mse}} = \frac{1}{N_\theta N_f} \sum_{i=1}^{N_\theta} \sum_{j=1}^{N_f} \left(\psi(x_j, y_j, \theta_i) - \psi_e(x_j, y_j, \theta_i)\right)^2.$$

1. Fixed shape parameter

First, the Gauss function with a fixed shape parameter $\varepsilon = 0.1$ was chosen as the basis function, and a polynomial degree of one was used. In Fig. (3), the numerical solution (for the direction) and the exact solution on $\theta = \pi$ are presented for a mesh of size 41×41. It is important to note that the vertical color bar on the left side of Fig. (3) represents the error of the numerical method. The color green indicates smaller errors, while darker red corresponds to larger errors. From Fig. (3), it is evident that the numerical solution exhibits a larger error in the boundary region compared to the interior, where the error is relatively small.
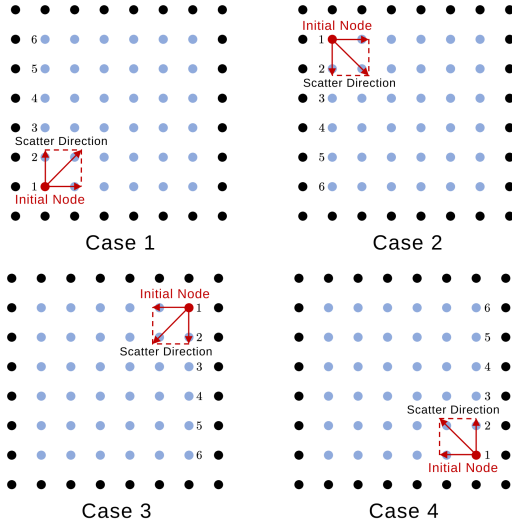
Fig. 4. Global nodal numbers

For the Gauss function with fixed shape parameters, different values of the shape parameter ε were tested multiple times to solve the neutron transport equation. By implementing a numerical solution in MATLAB, which iteratively computed the solution for each value of ε, error plots corresponding to two types of errors at four different resolutions were generated. In Fig. (5), the error between the numerical solution and the analytical solution is shown for various ε values. It can be observed that the smallest error between the numerical and analytical solutions occurred for $\varepsilon = 0.1$.

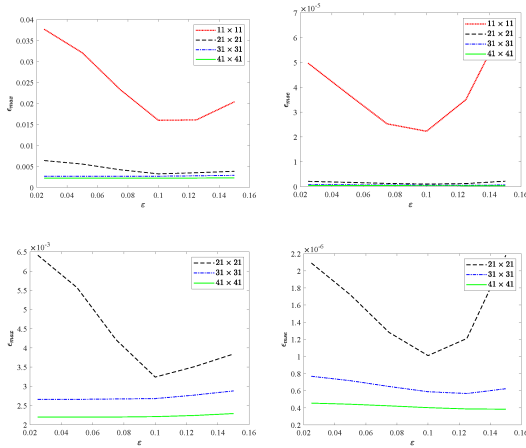Fig. 5. Errors vary with the shape parameter ε at four different resolutions of $N = 11 \times 11, 21 \times 21, 31 \times 31, 41 \times 41$.

The left side of Fig. (5) illustrates the variation of the maximum error $\epsilon_{\max}$ with respect to the shape parameter ε at four different resolutions: $N = 11 \times 11, 21 \times 21, 31 \times 31, 41 \times 41$. On the right side, the relationship between the shape parameter and the mean square error is shown for the same resolutions. At higher resolutions, the error did not exhibit significant variations with changes in the shape parameter due

to the chosen range of parameters, and the RBF-FD method achieved high accuracy. For a relatively coarse grid of 21×21 nodes, the maximum error was on the order of 10^{-3}, while the mean square error was on the order of 10^{-6}. Furthermore, for each resolution, there existed an optimal value for the shape parameter ε that minimized the error. However, for smaller values of ε, the resulting linear system became ill-posed, and rounding errors began to affect the solution's accuracy.

Next, the addition of a linear polynomial using the Gaussian basis function with a fixed shape parameter $\varepsilon = 0.1$ was examined. Figures 6 and 7 show the error distribution at various directions when the mesh size was 15×15. Specifically, the directions were discretized into 64 distinct angles, including $\theta = 0, \frac{\pi}{4}, \frac{\pi}{2}, \frac{3\pi}{4}, \pi, \frac{5\pi}{4}, \frac{3\pi}{2}, \frac{7\pi}{4}$. The figures illustrate the errors along these directions for the Gauss-FD method.

In Fig. 6, the errors are presented for directions corresponding to $\theta = 0, \frac{\pi}{2}, \pi, \frac{3\pi}{2}$, which represent the main axes (horizontal and vertical). In Fig. 7, the errors are shown for angles $\theta = \frac{\pi}{4}, \frac{3\pi}{4}, \frac{5\pi}{4}, \frac{7\pi}{4}$, which correspond to the diagonals. These visualizations help assess the accuracy of the Gauss-FD method across different directions. These results validate the accuracy and convergence of the proposed RBF-based method.
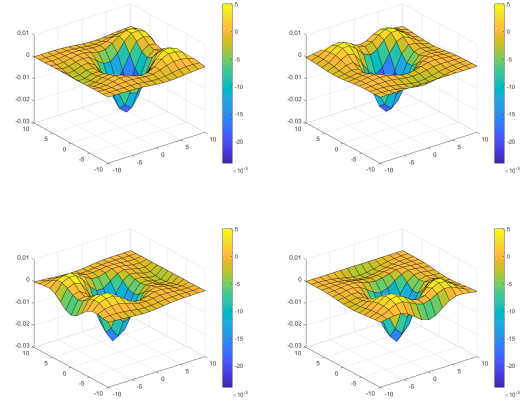

Fig. 6. Errors in four directions with angles $0, \frac{\pi}{2}, \pi, \frac{3\pi}{2}$ relative to the x-axis for the Gauss-FD method.

2. *Variable shape parameter*

Using the LOOCV method, the optimal shape parameter ϵ for the interpolation approximation of the unknown function ψ was determined for each stencil through calculation. The Gauss-FD method was then applied to solve the neutron transport equation. As shown in Fig. (8), when each stencil had a fixed ϵ value, larger errors were primarily concentrated near the boundaries of the region. However, when the LOOCV method was used to obtain distinct ϵ values for each stencil, the boundary errors were significantly reduced, although the error at the internal nodes downstream of the upwind mode increased noticeably. Furthermore, up to seven nodes were
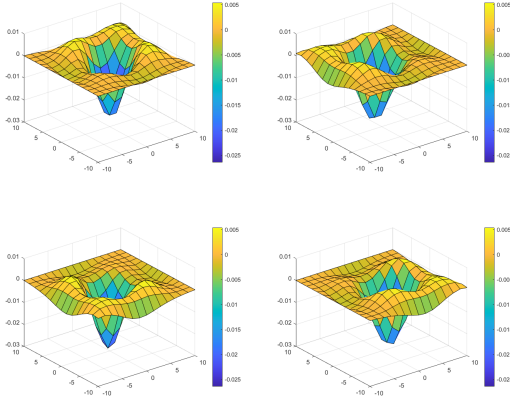
Fig. 7. Errors in four directions with angles $\frac{\pi}{4}$, $\frac{3\pi}{4}$, $\frac{5\pi}{4}$, $\frac{7\pi}{4}$ relative to the x-axis for the Gauss-FD method.
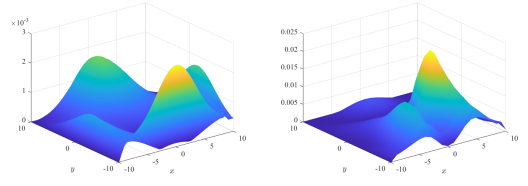
selected for each stencil, and the LOOCV method influenced the accuracy of the RBF-based shape parameter calculation.

The point-by-point error distribution of the numerical solution obtained using the RBF-FD method for both shape parameter types is presented in Fig. (9). It can be observed that a globally fixed shape parameter value improved numerical accuracy, but introduced multiple error peaks in the solution. In contrast, the shape parameters determined by the LOOCV method for each stencil minimized this issue, reducing the number of error peaks and enhancing the stability of the numerical method.
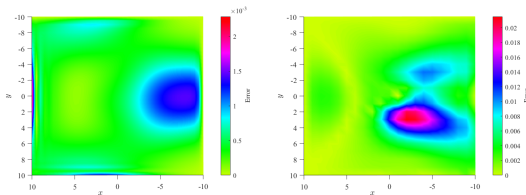

Fig. 8. Error distribution of the numerical results using the Gauss radial basis function with respect to shape parameters: (left) fixed parameter, (right) shape parameters obtained using the LOOCV method.

Fig. 9. Error distribution of the numerical results using the Gauss radial basis function for different shape parameter strategies: (left) fixed shape parameter, (right) shape parameters determined by the LOOCV method.

These results demonstrate the accuracy and convergence of the proposed finite-difference method based on radial basis functions (RBFs). However, challenges remain in solving the two-dimensional neutron transport equations with isotropic solutions. Specifically, because the solution is isotropic, the scattering terms calculated using numerical integration methods were accurate, with errors primarily arising from the spatial dependence of the solution. As shown in the left panel of Fig. (8), larger errors were concentrated near the boundaries of the two-dimensional space. This can be attributed to the susceptibility of RBFs to the Runge phenomenon, which tends to amplify errors near the boundary, causing larger inaccuracies in these regions compared to the interior. Nevertheless, by using shape parameters determined through the LOOCV method, the solution to the neutron transport equation exhibited significantly smaller errors, particularly in regions where the boundary was flat.

V.   CONCLUSION

This study introduces a RBF-FD for solving the neutron transport equation in two-dimensional space. Through detailed numerical analysis, we examine the error behavior and convergence properties of the proposed approach in capturing the spatial correlations inherent in the neutron transport equation. The results from numerical experiments validate the effectiveness of the method and demonstrate its suitability for accurately addressing the neutron transport equation.

The key advantages of the proposed method are as follows:

- The method achieves high accuracy with a relatively small number of nodes, ensuring computational efficiency.

- It is fully meshless, which simplifies the generation of computational nodes, even in irregular domains. This stands in contrast to traditional grid-based methods and eliminates the need for structured computational grids.

- The method is easily extendable to three-dimensional spaces, as RBFs depend solely on the Euclidean distance between nodes, meaning that the algorithm's complexity remains unaffected by the dimensionality of the problem.

These advantages suggest that the RBF-enhanced finite-difference method is a promising approach for solving the neutron transport equation numerically. In future research, we plan to extend the RBF-FD method to tackle neutron transport equations in more complex physical domains, including those in three-dimensional space. This extension aims to enhance the method's applicability and accuracy in a broader range of real-world scenarios.

[1] E. E. Lewis, W. F. Miller Jr., Computational Methods of Neutron Transport, Am. Nucl. Soc., Lagrange Park, IL, 1993.

[2] Z. S. Xie, L. Deng, Neutron Transport Theory and Numerical Calculation Methods (Xi'an Jiaotong Univ. Press, Xi'an, 2022).**(in Chinese)**

[3] W. Wang, H. X. Zhang, X. X. Jiang et al., A high-order and efficient numerical technique for the nonlocal neutron diffusion equation representing neutron transport in a nuclear reactor, Ann. Nucl. Energy. **195**, 110163 (2024). https://doi.org/10.1016/j.anucene.2023.110163

[4] W. F. Liang, D. X. Qiu, W. L. Xiang et al., Enhancement of neutron irradiation uniformity for the CFBR-II fast burst reactor with a biaxial rotational technique, Nucl. Sci. Tech. **26**(2), 020201 (2015). https://doi.org/10.13538/j.1001-8042/nst.26.020201

[5] P. Cao, H. Ding, C. L. Cao et al., Study on the off situ reconstruction of the core neutron field based on dual-task hybrid network architecture, Nucl. Sci. Tech. **36**(8), (2025). https://doi.org/10.1007/s41365-024-01571-6

[6] Z. M. Duan, J. Zhang, G. H. Su et al., Optimization analysis of nuclear thermal coupling for a small nuclear thermal propulsion (SNTP) reactor, Nucl. Eng. Design. **417**, 112838 (2024). https://doi.org/10.1016/j.nucengdes.2023.112838

[7] Z. W. Liu, F. K. Roepke, Z. W. Han, Type Ia Supernova Explosions in Binary Systems: A Review, Res. Astron. Astrophys. **23**, 082001 (2023). https://doi.org/10.1088/1674-4527/acd89e

[8] A. Li, G. C. Yong, Y. X. Zhang, Testing the phase transition parameters inside neutron stars with the production of protons and lambdas in relativistic heavy-ion collisions, Phys. Rev. D. **107**(4), 043005 (2023). https://doi.org/10.1103/PhysRevD.107.043005.

[9] J. K. Shultis, R. E. Faw, Radiation Shielding. In: Meyers, R. (eds) Encyclopedia of Sustainability Science and Technology. Springer, New York, NY. (2016). https://doi.org/10.1007/978-1-4939-2493-6_25-5

[10] X.-Y. Wang, J.-Y. Chen, Q. Zhang, Boron shielding design for neutron and gamma detectors of a pulsed neutron tool, Nucl. Sci. Tech. **35**, 1-10 (2024). https://doi.org/10.1007/s41365-023-01359-0

[11] B. Liu, F. Gao, H. Chen, Development of plasma-facing materials for fusion reactors, Nucl. Sci. Tech. **35**, 20-30 (2024). https://doi.org/10.1007/s41365-024-01573-4

[12] X. Wang, Y. Li, Z. Zhang, Advances in magnetic confinement fusion research in China, Nucl. Sci. Tech. **35**, 1-15 (2024). https://doi.org/10.1007/s41365-024-01573-4

[13] J. R. Lamarsh, Introduction to nuclear reactor theory, Nucl. Sci. Tech. **26**, s20503 (1966), Addison-Wesley Publication Company, Massachusetts. https://doi.org/10.1063/1.3034192

[14] W. Tong, M. Xu, H. Li, The effects of neutron irradiation on the electrical characteristics of high-power thyristor, Nucl. Sci. Tech. **34**, 134 (2023). https://doi.org/10.1007/s41365-023-01290-4

[15] Y. Yan, et al., High-temperature steam oxidation study of irradiated FeCrAl defueled specimens, J. Nucl. Mater. **590**, 134 (2024). https://doi.org/10.1016/j.jnucmat.2023.154868

[16] H.H. Chi, Y.H. Wang, Y. Ma, Reduced-order with least square-finite difference method for neutron transport equation, Ann. Nucl. Energy. **191**, 109914 (2023). https://doi.org/10.1016/j.anucene.2023.109914

[17] K. Assogba, L. Bourhrara, I. Zmijarevic et al., Spherical Harmonics and Discontinuous Galerkin Finite Element Methods for the Three-Dimensional Neutron Transport Equation: Application to Core and Lattice Calculation. Nucl. Sci. Eng. **197**(8), 1584-1599 (2023). https://doi.org/10.1080/00295639.2022.2154546

[18] G. Zhang, H. Zhang, TARS: A parallel tetrahedral discontinuous finite element code for the solution of the discrete ordinates neutron transport equation. Ann. Nucl. Energy. **196**, 110242 (2024). https://doi.org/10.1016/j.anucene.2023.110242

[19] S.D. Li, et al., Angular-spatial discontinuous Galerkin method for radiative heat transfer with a participating medium in complex three-dimensional geometries. Int. Commun. Heat Mass Transfer. **145**, 106836 (2023). https://doi.org/10.1016/j.icheatmasstransfer.2023.106836

[20] X.Y. Li, et al., Research on neutron diffusion equation and nuclear thermal coupling method based on gradient updating finite volume method, Ann. Nucl. Energy. **195**, 110158 (2024). https://doi.org/10.1016/j.anucene.2023.110158

[21] A.M.G. Cox, S.C. Harris, A.E. Kyprianou, M. Wang, Monte Carlo Methods for the Neutron Transport Equation, SIAM/ASA J. Uncertainty Quantification. **10**(2), 775-825 (2022). https://doi.org/10.1137/21M1390578

[22] M. Dai, M.S. Cheng, A low order MOC-based synthetic acceleration scheme of the MOC neutron transport method for molten salt reactors, Ann. Nucl. Energy. **208**, 110789 (2024). https://doi.org/10.1016/j.anucene.2024.110789

[23] A. Barbarino, Numerical Methods for Neutron Transport Calculations of Nuclear Reactors, 2014.

[24] T. Tanbay, B. Ozgener, Fully meshless solution of the one-dimensional multigroup neutron transport equation with the radial basis function collocation method, Comput. Math. Appl. **79**(5), 1266-1286 (2020). https://doi.org/10.1016/j.camwa.2019.08.037

[25] S. Tayefi, A. Pazirandeh, M.A. Kheradmand Saadi, A meshless local Petrov–Galerkin method for solving the neutron diffusion equation, Nucl. Sci. Tech. **29**, 169 (2018). https://doi.org/10.1007/s41365-018-0506-x

[26] G. R. Liu, G. Liu, Meshfree methods with MATLAB, CRC Press (2009). https://doi.org/10.1201/9781420082104.

[27] G. E. Fasshatier, J. G. Zhang, On choosing "optimal" shape parameters for RBF approximation, Numer. Algorithms. **45**, 345-368 (2007). https://doi.org/10.1007/s11075-007-9072-8.

[28] V. Bayona, M. Moscoso, M. Carretero,et al., RBF-FD formulas and convergence properties, J. Comput. Phys. **229**, 8281-8295 (2010). https://doi.org/10.1016/j.jcp.2010.07.008.

[29] E. Larsson, B. Fornberg, Theoretical and computational aspects of multivariate interpolation with increasingly flat radial basis functions, Comput. Math. Appl. **49**, 103–130 (2005). https://doi.org/10.1016/j.camwa.2005.01.010.

[30] A. Muñoz Oliva, H. Alves Filho, D. J. M. e Silva, C. R. García Hernández, The spectral nodal method applied to multigroup SN neutron transport problems in One-Dimensional geometry with Fixed–Source, Prog. Nucl. Energy. **105**, 106–113 (2018). https://doi.org/10.1016/j.pnucene.2017.12.017.

[31] J. M. Zhao, J. Y. Tan, L. H. Liu, A second order radiative transfer equation and its solution by meshless method with application to strongly inhomogeneous media, J. Comput. Phys. **232**, 431–455 (2013). https://doi.org/10.1016/j.jcp.2012.08.020.

[32] N. Flyer, B. Fornberg, V. Bayona, G. A. Barnett, On the role of polynomials in RBF-FD approximations: I. Interpolation and accuracy, J. Comput. Phys. **321**, 21–38 (2016). https://doi.org/10.1016/j.jcp.2016.05.026.

[33] V. Bayona, N. Flyer, B. Fornberg, G. A. Barnett, On the role of polynomials in RBF-FD approximations: II. Numerical solution of elliptic PDEs, J. Comput. Phys. **332**, 257–273 (2017). https://doi.org/10.1016/j.jcp.2016.12.008.

[34] V. Bayona, N. Flyer, B. Fornberg, On the role of polynomials in RBF-FD approximations: III. Behavior near domain boundaries, J. Comput. Phys. **380**, 378–399 (2019). https://doi.org/10.1016/j.jcp.2018.12.013.

[35] G. A. Barnett, N. Flyer, L. J. Wicker, An RBF-FD polynomial method based on polyharmonic splines for the Navier-Stokes equations: Comparisons on different node layouts, J. Comput. Phys. (2015). https://doi.org/10.48550/arXiv.1509.02615.

[36] J. Hu, Positivity-preserving acceleration algorithm for neutron transport equation. Master's thesis, Jiangxi Normal University (2020). (in Chinese)

[37] B. Yang, Discrete ordinate method is used to solve the transport equation with anisotropic scattering. Master's thesis, China Academy of Engineering Physics (2005). (in Chinese)

[38] M. L. Adams, E. W. Larsen, Fast iterative methods for discrete-ordinates particle transport calculations, Prog. Nucl. Energy, **40**, 3–159 (2002). https://doi.org/10.1016/S0149-1970(01)00023-3.